We claim:

1. At a near end of a transmission system, a method of mapping client data into an outgoing block, said outgoing block comprising payload words and overhead words, so as to transfer said client data and clock synchronization information associated with said client data to a far end of said transmission system, said method comprising:

receiving client words of an incoming flow of said client data;

generating fill words;

mapping said payload words into said outgoing block, where:

each of a first plurality of said payload words comprises one of said client words; and

each of a second plurality of said payload words comprises one of said fill words;

adjusting a rate at which said client words are mapped such that an average number of said client words mapped is equal to an average number of said client words received;

including an indication of said rate in said overhead words; and

mapping said overhead words into said outgoing block.

2. The method of claim 1 wherein said indication of said rate comprises a first value (YNOM) representative of a nominal magnitude of said rate and an indicator (YOFF) of an offset from said nominal magnitude.

3. The method of claim 2 further comprising:

detecting a difference between a number of said client words received and a number of said client words mapped; and

converting said difference to said offset indicator.

4. The method of claim 3 further comprising, before said converting, filtering said difference and utilizing said filtered difference to optimize said adjusting of said rate.

5. The method of claim 2 further comprising generating a first timing signal, where said mapping said payload words into said outgoing block comprises mapping one of said first plurality of said payload words as triggered by said first timing signal.

6. The method of claim 5 wherein said adjusting said rate at which said client words are mapped comprises adjusting said first timing signal.

7. The method of claim 5 further comprising:

while mapping said payload words into said outgoing block, increasing an accumulated sum in an accumulator, at each tick of a near end system clock, by a sum of said offset indicator and said nominal magnitude of said rate; and

responsive to mapping one of said payload words and where said accumulated sum is greater than a pre-set amount, decreasing said accumulated sum by said pre-set amount.

8. The method of claim 7 further comprising, while mapping said overhead words into said outgoing block, increasing said accumulated sum in said accumulator, at each said tick of said far end system clock, by said sum of said offset indicator and said nominal magnitude of said rate.

9. The method of claim 7 further comprising:

responsive to said decreasing said accumulated sum, generating a flag; and

deriving said first timing signal from said flag.

10. The method of claim 7 further comprising:

determining a synchronization (sync) indicator, at a given point relative to a beginning of a data structure for said outgoing block, for use in synchronizing a second timing signal, at said far end of said transmission system, with said first timing signal; and

including said sync indicator in one of said overhead words.

11. The method of claim 10 further comprising sampling, at said point relative to said beginning of said data structure, said accumulated sum in said accumulator.

12. The method of claim 11 wherein said data structure is a SONET synchronous payload envelope.

13. The method of claim 2 further comprising storing said first value representative of said nominal magnitude of said rate.

14. The method of claim 2 further comprising:

recovering a clock signal from said incoming flow of client data;

converting said incoming flow of client data from serial bits to said client words; and

converting said clock signal to a parallel clock signal.

15. The method of claim 1 further comprising including error correction and detection information in said overhead words.

16. A mapping processor comprising:

means for receiving client words of an incoming flow of client data;

means for generating fill words;

means for mapping payload words into an outgoing block, where:

each of a first plurality of said payload words comprises one of said client words; and

each of a second plurality of said payload words comprises one of said fill words;

means for adjusting a rate at which said client words are mapped such that an average number of said client words mapped is equal to an average number of said client words received;

means for including an indication of said rate in a plurality of overhead words; and

means for mapping said plurality of overhead words into said outgoing block.

17. A mapping processor comprising:

a receiver for receiving client words of an incoming flow of client data;

a fill unit for generating fill words;

a mapping unit for:

adjusting a rate at which said client words are mapped such that an average number of said client words mapped is equal to an average number of said client words received;

including an indication of said rate in a plurality of overhead words;

mapping payload words into an outgoing block, where:

each of a first plurality of said payload words comprises one of said client words;

each of a second plurality of said payload words comprises one of said fill words; and

mapping said plurality of overhead words into said outgoing block.

18. A mapping processor operable to:

receive client words of an incoming flow of client data;

generate fill words;

map payload words into an outgoing block, where:

each of a first plurality of said payload words comprises one of said client words; and

each of a second plurality of said payload words comprises one of said fill words;

adjust a rate at which said client words are mapped such that an average number of said client words mapped is equal to an average number of said client words received;

include an indication of said rate in a plurality of overhead words; and

map said plurality of overhead words into said outgoing block.

19. A computer readable medium for providing program control to a mapping processor, said computer readable medium adapting said processor to be operable to:

receive client words of an incoming flow of client data;

generate fill words;

map payload words into an outgoing block, where:

each of a first plurality of said payload words comprises one of said client words; and

each of a second plurality of said payload words comprises one of said fill words;

adjust a rate at which said client words are mapped such that an average number of said client words mapped is equal to an average number of said client words received;

include an indication of said rate in a plurality of overhead words; and

map said plurality of overhead words into said outgoing block.

20. A method of regenerating a clock associated with a tributary signal, said method comprising:

generating a local synchronization signal;

receiving a block of words, said block comprising overhead words and payload words;

de-mapping, from said block, an overhead word including an offset indicator;

altering said generating using said offset indicator to obtain an altered local synchronization signal; and

synthesizing said clock associated with said tributary signal using said altered local synchronization signal.

21. The method of claim 20 further comprising de-mapping, from said block, an overhead word including a value related to a nominal frequency of a timing signal and wherein said generating said local synchronization signal further comprises:

while de-mapping said payload words from said block, increasing an accumulated sum in an accumulator, at each tick of a far end system clock, by a sum of said offset indicator and said nominal frequency value; and

responsive to de-mapping a payload word and where said accumulated sum is greater than a pre-set amount, decreasing said accumulated sum by said pre-set amount.

22. The method of claim 21 wherein said generating said local synchronization signal further comprises, while de-mapping said overhead words from said block, increasing said accumulated sum in said accumulator, at each said tick of said far end system clock, by said sum of said offset indicator and said nominal frequency value.

23. The method of claim 21 wherein synthesizing said synthesized clock comprises:

converting said altered local synchronization signal to a converted synchronization signal; and

deriving said synthesized clock from said converted synchronization signal;

where said converted synchronization signal has a frequency based on a rate at which said accumulated sum is decreased by said pre-set amount.

24. The method of claim 21 further comprising de-mapping overhead words including nominal frequency values until consecutively de-mapped nominal frequency values are equal before said generating a local synchronization signal using said nominal frequency value.

25. The method of claim 20 further comprising:

generating a timing signal; and

de-mapping, from said block, a payload word when triggered by said timing signal, where said payload word includes a word of client data.

26. The method of claim 25 further comprising generating a flag and deriving said timing signal from said flag.

27. The method of claim 26 wherein said flag is generated in response to an accumulator exceeding a predetermined sum and said timing signal is a gapped clock with a rising edge triggered by said flag.

28. The method of claim 25 further comprising converting said word of client data to a serial stream of bits, where said stream of bits has timing derived from said synthesized clock.

29. The method of claim 20 further comprising:

de-mapping, from said block, an overhead word including a synchronization (sync) indicator;

determining a local sync indicator from said altered local synchronization signal corresponding to said de-mapped sync indicator;

comparing said local sync indicator with said de-mapped sync indicator to result in a difference value; and

altering said generating based upon said difference value.

30. The method of claim 29 further comprising, before said altering said generating based upon said difference value, repeating said de-mapping said overhead word including said sync indicator, said determining and said comparing, where said repeating results in a plurality of difference values and said repeating is performed until consecutive ones of said plurality of difference values are equal.

31. The method of claim 20 further comprising, before said synthesizing, filtering said altered local synchronization signal.

32. The method of claim 20 wherein said synthesizing comprises:

generating an approximation of said clock from said synchronization signal;

detecting a difference between said clock and said synchronization signal;

converting said difference to a frequency offset; and

altering said generating said clock by said frequency offset, where said altering serves to minimize said difference.

33. The method of claim 20 further comprising,

receiving an indication of a pointer adjustment;

responsive to said receiving said pointer adjustment, adding a predetermined value to said altered local synchronization signal; and

repeating said adding while progressively reducing a magnitude of said predetermined value until said magnitude is zero.

34. A de-mapping processor comprising:

means for generating a local synchronization signal;

means for receiving a block of words, said block comprising overhead words and payload words;

means for de-mapping, from said block, an overhead word including an offset indicator;

means for altering said generating using said offset indicator to obtain an altered local synchronization signal; and

means for synthesizing said clock associated with said tributary signal using said altered local synchronization signal.

35. A de-mapping processor comprising:

a receiver for receiving a block of words, said block comprising overhead words and payload words;

a de-mapper for de-mapping, from said block, an overhead word including an offset indicator;

a signal source for generating a local synchronization signal where said signal source allows for altering said generating using said offset indicator to obtain an altered local synchronization signal; and

a synthesizer for synthesizing said clock associated with said tributary signal using said altered local synchronization signal.

36. The de-mapping processor of claim 35 wherein said synthesizer comprises an accumulator, a sine lookup circuit, a quantization filter and a limitter.

37. A de-mapping processor operable to:

receive a block of words, said block comprising overhead words and payload words;

de-map, from said block, an overhead word including an offset indicator;

generate a local synchronization signal;

alter said local synchronization signal using said offset indicator to obtain an altered local synchronization signal; and

synthesize said clock associated with said tributary signal using said altered local synchronization signal.

38. A computer readable medium for providing program control to a de-mapping processor, said computer readable medium adapting said processor to be operable to:

receive a block of words, said block comprising overhead words and payload words;

de-map, from said block, an overhead word including an offset indicator;

generate a local synchronization signal;

alter said local synchronization signal using said offset indicator to obtain an altered local synchronization signal; and

synthesize said clock associated with said tributary signal using said altered local synchronization signal.

39. A computer data signal embodied in a carrier wave comprising;

words from a flow of data received at a mapping processor; and

an offset indicator generated at said mapping processor.